

Domain: 192.168.0.109

Vulnerability Name	Sensitive information in the URL
Severity	8.0 High
CVSS Score	Base Score: 8.0 High CVSS:3.1/AV:A/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H
Description	The "Sensitive information in the URL" vulnerability refers to the practice of including sensitive data, such as authentication credentials, API keys, or other confidential information, directly in the URL of a web application. Observation: While testing the application we have observed that the application’s login operation was implemented in GET method and username and passwords are exposed in the URL.
Attack Scenario	While testing the application we have observed that the application’s login operation was implemented in GET method and username and passwords are exposed in the URL. So As an attacker we have performed MITM attack in the local network and able to capture the victim user’s login credentials: Impact: This can lead to security risks as URLs are often logged in various places, such as web server logs, browser history, and referer headers, potentially exposing sensitive information.
Vulnerable Location	http://192.168.0.109/dvwa/vulnerabilities/brute/?username=admin&password=12345&Login=Login
Vulnerable Parameters	Vulnerable operation: Login operation Parameters: username & password
Reproduction Steps	1. Access the login page URL. 2. URL: http://192.168.0.109/dvwa/vulnerabilities/brute/ 3. Enter username and password, click on the login button. 4. Observe that the username and password in the URL. 5. Also recorded in the browser history.
Remediation	1. Use HTTPS: Always use HTTPS to encrypt data transmitted between the client and server, including URLs. This helps protect sensitive information from being intercepted during transit. 2. Avoid Including Sensitive Data in URLs: Refrain from including sensitive information, such as passwords or API keys, directly in the URL. Instead, use secure methods for transmitting sensitive data, such as POST requests with form data.
Sample Code	N/A
Reference	Link1 Link2

Proof of concept:

1	http://192.168.0.109	GET	/dvwa/vulnerabilities/brute/?us...	✓	200	4885	HTML	Damn Vulnerable W...
2	http://192.168.0.109	GET	/dvwa/vulnerabilities/brute/?us...	✓	200	4885	HTML	Damn Vulnerable W...
33	http://192.168.0.109	GET	/dvwa/vulnerabilities/brute/?us...	✓	200	4951	HTML	Damn Vulnerable W...

Request

PrettyRaw\nActions

1GET /dvwa/vulnerabilities/brute/?username=admin&password=12345&Login=Login HTTP/1.1

2Host: 192.168.0.109

3User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:120.0) Gecko/20100101 Firefox/120.0

4Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8

5Accept-Language: en-US,en;q=0.5

6Accept-Encoding: gzip, deflate

7Connection: close

8Referer: http://192.168.0.109/dvwa/vulnerabilities/brute/?username=admin&password=12345&Login=Login

9Cookie: security=high; PHPSESSID=fb4a152aff52180701c62995e545342a

10Upgrade-Insecure-Requests: 1

11

12

Response

PrettyRawRender\nActions

1HTTP/1.1 200 OK

2Date: Tue, 12 Dec 2023 17:21:09 GMT

3Server: Apache/2.2.8 (Ubuntu) DAV/2

4X-Powered-By: PHP/5.2.4-2ubuntu5.10

5Pragma: no-cache

6Cache-Control: no-cache, must-revalidate

7Expires: Tue, 23 Jun 2009 12:00:00 GMT

8Connection: close

9Content-Type: text/html; charset=utf-8

10Content-Length: 4575

11

12

13<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//

14

15<html xmlns="http://www.w3.org/1999/xhtml">

16

17<head>

18<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

19

20<title>

21Damn Vulnerable Web App (DVWA) v1.0.7 :: Vulnerability Demo

22</title>

23

24<link rel="stylesheet" type="text/css" href="css/brute.css">

25<link rel="icon" type="image/ico" href="img/favicon.ico">

0 matches

0 matches

INSPECTOR